Aalto University
School of Science
Master's Programme in Computer, Communication and
Information Sciences

Roope Palomäki

A distance-aware 2D barcode for mobile computing applications

Master's Thesis
Espoo, Oct 22, 2018

Supervisor and instructor: Mario Di Francesco, Associate Professor

Aalto University
School of Science
Master's Programme in Computer,
Communication and Information Sciences

ABSTRACT OF
MASTER'S THESIS

Author:	Roope Palomäki		
Title:	A distance-aware 2D barcode for mobile computing applications		
Date:	October 22, 2018	**Pages:**	49
Major:	Mobile Computing, Services and Security	**Code:**	SCI3045
Supervisor:	Mario Di Francesco, Associate Professor		
Instructor:	Mario Di Francesco, Associate Professor		

Global internet use is becoming increasingly mobile, and mobile data usage is growing exponentially. This puts increasing stress on the radio frequency spectrum that cellular and Wi-Fi networks use. As a consequence, research has also been conducted to develop wireless technologies for other parts of the electromagnetic spectrum – namely, visible light.

One approach of using the visible light channel for wireless communication leverages barcodes. In this thesis, we propose a 2D barcode that can display different information based on the distance between the barcode and the scanner. Earlier research on distance-sensitive barcodes has focused on providing a closer viewer more information as a closer viewer can see more detail. In contrast, we target use cases where a clear physical separation between users of different roles can be made, such as presentation systems.

We evaluate two methods of achieving distance-awareness: color-shifting of individual colors, where a color changes tone at longer distances, and color blending, where two colors blend into a third color at longer viewing distances. Our results show that a modern smartphone is capable of leveraging color-shifting in ideal conditions, but external changes such as ambient lighting render color-shifting unusable in practical scenarios. On the other hand, color blending is robust in varying indoor conditions and can be used to construct a reliable distance-aware barcode.

Accordingly, we employ color blending to design a distance-aware barcode. We implement our solution in an off-the-shelf Android smartphone. Experimental results show that our scheme achieves a clear separation between close and far viewers. As a representative use case, we also implement a presentation system where a single barcode provides the presenter access to presentation tools and the audience access to auxiliary presentation material.

Keywords:	wireless communication, camera-display communication, barcode, distance-awareness, smartphone, mobile computing
Language:	English

Aalto-yliopisto
Perustieteiden korkeakoulu
Tieto-, tietoliikenne- ja informaatiotekniikan maisteriohjelma

DIPLOMITYÖN TIIVISTELMÄ

Tekijä:	Roope Palomäki		
Työn nimi:	Etäisyyden huomioiva kaksiulotteinen viivakoodi mobiilikäyttötapauksiin		
Päiväys:	22. lokakuuta 2018	**Sivumäärä:**	49
Pääaine:	Mobile Computing, Services and Security	**Koodi:**	SCI3045
Valvoja:	Mario Di Francesco, Associate Professor		
Ohjaaja:	Mario Di Francesco, Associate Professor		

Maailmanlaajuinen internetin käyttö muuttuu yhä liikkuvammaksi, ja mobiilidatan käyttö kasvaa eksponentiaalisesti. Tämä kohdistaa yhä suurempia vaatimuksia radiotaajuusspektriin, jota mobiili- ja Wi-Fi-verkot käyttävät. Näin ollen tutkijat ovat kehittäneet langattomia teknologioita hyödyntäen myös muita sähkömagneettisen spektrin osia – erityisesti näkyvää valoa.

Yksi näkyvän valon sovellus langattomassa viestinnässä ovat viivakoodit. Tässä työssä kehitämme kaksiulotteisen viivakoodin, joka pystyy välittämään eri tietoa katselijoille eri etäisyyksillä. Aiempi etäisyyden huomioivien viivakoodien tutkimus on keskittynyt tarjoamaan lähellä olevalle katselijalle enemmän tietoa, koska läheinen katselija näkee viivakoodin tarkemmin. Sitä vastoin me keskitymme käyttötapauksiin, joissa eri käyttäjäroolien välillä on selkeä etäisyydellinen ero, kuten esimerkiksi esitelmissä puhujan ja yleisön välillä.

Tarkastelemme kahta menetelmää: yksittäisten värien muutoksia etäisyyden muuttuessa ja kahden värin sekoittumista etäisyyden kasvaessa. Tulostemme perusteella nykyaikainen älypuhelin pystyy hyödyntämään yksittäisten värien muutoksia ihanteellisissa olosuhteissa, mutta ulkoiset tekijät, kuten ympäristön valaistus, aiheuttavat liian suuria värimuutoksia käytännön käyttötapauksissa. Toisaalta värien sekoittuminen on johdonmukaista muuttuvassa sisäympäristössä ja sitä voidaan käyttää luotettavan viivakoodin luomisessa.

Näin ollen me suunnittelemme etäisyyden huomioivan viivakoodin hyödyntäen värien sekoittumista. Toteutamme ratkaisumme yleisesti saatavilla olevalle Android-älypuhelimelle. Kokeellisten tulostemme perusteella menetelmämme saavuttaa selkeän erottelun läheisten ja kaukaisten katselijoiden välillä. Esimerkkikäyttötapauksena toteutamme myös esitelmäjärjestelmän, jossa sama viivakoodi antaa lähellä olevalle puhujalle nopean pääsyn esitystyökaluihin ja kauempana olevalle yleisölle pääsyn esityksen apumateriaaliin.

Asiasanat:	langaton viestintä, näkyvä valo, viivakoodi, etäisyys, älypuhelin, mobiilitekniikka
Kieli:	englanti

Table of Contents

1 Introduction

Global internet use is becoming increasingly mobile. In 2018, the total number of active mobile internet users has climbed over 3.8 billion, over 90 percent of the total number of active internet users and approximately half of the total human population [1]. Over half of the total web traffic comes from mobile phones. The total monthly global mobile data traffic has reached nearly 15 exabytes (billions of gigabytes), growing by at least 50 percent annually since 2013. Ericsson forecasts a compound annual growth rate of 43 percent, leading to almost 107 exabytes monthly by the end of 2023, nearly eight times the current consumption [2].

Together with the exponentially increasing mobile data use, Wi-Fi and other wireless networks are becoming increasingly ubiquitous. Cisco forecasts that the total number of Wi-Fi hotspots will increase almost six-fold from 94 million in 2016 to 541.6 million by 2021 [3]. This intensifies stress on the already urgently-in-demand radio frequency spectrum used by these technologies, leading to new ranges of the frequency band being opened for unlicensed use [4].

However, the radio and microwave frequencies are a limited, regulated state resource; managing congestion has become a major concern [5, 6, 7, 8]. Consequently, researchers have also turned to other parts of the electromagnetic spectrum for communication purposes, notably the visible spectrum (also referred to as visible light). Having several orders of magnitude more spectrum than the radio band [9], the visible light band offers great theoretical bandwidth potential. Being physically contained by objects such as walls, it remains unlicensed and regulatorily simple. Wireless communication systems using the visible spectrum are often called visible light communication (VLC) systems.

Given the virtual omnipresence of cameras in smartphones, VLC provides great opportunities for wireless communication in mobile computing applications. Especially barcodes have proven an effective way of communicating limited amounts of information. Barcodes require little setup and few resources: a transmitter can be as inexpensive as a piece of paper that a code is printed on, and virtually any device with a camera can act as a scanner. Combined with advances in computer vision,

research in optical camera communication (OCC) has led to inventions such as the ubiquitous QR code [10, 11], originally developed for the automotive industry in 1994 [12]. A multitude of other barcode designs with differing properties have been developed to cover various use cases. These will be further discussed in Chapter 2.

1.1 Research topic and goals

While data consumption and users are becoming increasingly mobile, there is relatively little research on mobile VLC. In the context of this thesis, by "mobile VLC" we mean that the receiver is mobile in relation to the transmitter, i.e., the distance between the two may vary. Most of the existing literature focuses on the traditional VLC approach of LED transmitters being observed by custom optical receivers measuring the power received [13, 14, 15, 8]. However, only some consider the effect of mobility, or distance, on scenarios suitable for use with general-purpose digital displays, cameras, and smart devices. Most relevantly, [16] and [17] consider encoding information in different levels of spatial detail in barcode-like codes to provide better reach and to deal with common channel impairments.

The existing literature has considered distance mainly due to signal degradation as distance grows, essentially aiming at ensuring the receipt of critical information at longer distances. The works leveraging Fourier and Haar transforms to encode data in the frequency domain generally provide a somewhat continuous spectrum of outputs depending on channel quality. On the other hand, the motivation of this thesis stems from use cases where a clear separation between a close and a distant observer can be made. Such use cases include, for example, applications in presentation systems, where the presenter and audience are clearly separate entities at viewing distances that can clearly be classified as near and far.

The scope of the thesis is limited to methods that can be utilized in mobile computing applications. Thus, the receiver used is a common smartphone camera, and transmitters used are smartphone screens and other common digital displays. Furthermore, we aim at using color-based methods as they provide higher data capacity than systems based on a single LED and are inherently suitable for digital displays. In addition, we intend to create a practical demonstration of a sample use case, and

choose the presentation system mentioned above as a representative scenario.

Consequently, we define the goals of this thesis as follows:

1. *Define a two-dimensional distance-aware barcode that creates different output zones based on viewing distance and achieves a clear separation between a close viewer and a distant viewer.*
2. *Assess the performance of the system in common practical scenarios.*
3. *Develop a proof-of-concept application and showcase the system in a common potential use case.*

1.2 Structure

The thesis is organized as follows. Chapter 2 introduces the background and details relevant literature. Chapter 3 develops and evaluates the two methods of designing a code considered in the scope of this thesis: color shifting of individual colors, and color blending of color combinations. Chapter 4 presents the proof-of-concept application and an example use case. Finally, Chapter 5 discusses the results and Chapter 6 concludes the thesis.

2 Background

This chapter provides an overview of optical wireless communication and the relevant subareas in order of specificity – visual light communication, optical camera communication, and screen-camera communication. Furthermore, it details existing research on barcodes and their uses for wireless communication.

2.1 Optical wireless communication

Optical wireless communication (OWC) considers the use of the infrared (IR), visible light (VL), and ultraviolet (UV) frequencies of the electromagnetic spectrum for wireless communication. These three parts of the spectrum, ranging approximately from 300 GHz to 30 PHz, represent a tremendously greater bandwidth than the radio frequencies, ranging from 3 kHz to 300 GHz.

Radio frequencies (RF) have historically formed the most widely used frequency band for communication purposes [8], especially due to the impressive physical distances lower frequencies can reach while remaining robust in our environment. However, the radio band also has several disadvantages compared to the IR, VL, and UV frequencies. Most importantly, radio frequencies are rapidly being depleted by the modern society, leading to strict regulation and licensing. Consequently, especially the parts of the radio band marked for unlicensed use are becoming increasingly congested, and managing the spectrum is a concern [6, 7, 18]. This congestion further leads to interference in common scenarios such as several physically co-located networks. The best example of this issue are the numerous, sometimes dozens, Wi-Fi networks operating over the same physical area. On the other hand, the optical spectrum considered in OWC provides several orders of magnitude more bandwidth and, excluding abnormally high power levels, the waves are generally physically contained by obstacles such as walls. Thus, congestion and interference are smaller concerns. In addition to lesser interference, physical containment also creates desirable safety properties as the signal is limited by the line of sight. In some use cases using RF technology can also be

considerably more expensive: the authors in [8] note that RF modules are approximately 5 times as expensive as modules with similar data transfer capabilities using visual light.

The first known use of optical wireless communication is the photophone by Alexander Graham Bell in 1880 [19], used to transmit audio with beams of sunlight over 200 meters. However, due to inadequate optical equipment, OWC research was not further developed for almost a century [20]. In the 1980s and 1990s, infrared technology was developed, eventually standardized by the IrDa (Infrared Data Association), and popularized in consumer use cases [21, 22, 23].

Technology using only the visible part of the electromagnetic spectrum, approximately 400 THz to 800 THz, for wireless communication is called visible light communication (VLC). Most VLC research considers LEDs as transmitters [20, 24]. LEDs are capable of switching on and off very rapidly, crucially fast enough for the human eye to perceive constant illumination instead of flickering. In addition, the number of LEDs used in indoor and outdoor lighting is rapidly increasing. The combination of these factors means that VLC has high potential and several use cases combined with lighting without affecting the main function of the lighting. Often suggested use cases include intelligent transport systems (ITS) utilizing roadside lights and signage and vehicle-to-vehicle (V2V) or vehicle-to-infrastructure (V2I), indoor localization and visual light positioning (VLP), and scenarios where electromagnetic interference (EMI) causes issues, such as hospitals and space applications.

VLC using fast-switching LEDs was first proposed by researchers in Japan [25, 26], eventually leading to the formation of the VLCC (Visible Light Communications Consortium) in 2003; its current successor is the VLCA (Visible Light Communications Association). Several standards have been proposed [27], including the IEEE 802.15.7 [28] which defines communication protocols for VLC with data rates up to 96 Mbps. It was later augmented with revision 802.15.7r1 [29] which includes optical camera communication, discussed next.

While VLC systems typically use white LEDs as transmitters and photodiodes (PD) as receivers [24, 20], research in optical camera communication (OCC) specifically proposes using cameras or image sensors as receivers, allowing for easier deployment on current smart devices. Whereas VLC based on LEDs and PDs is limited by LED

technology, OCC is generally limited by camera technology, especially the frame rate: common smart device cameras are often only capable of capturing 30 frames per second, thereby restricting the achievable data rate. The sensor technology type, CMOS or CCD, also affects OCC systems. Most importantly, CCD sensors capture every pixel concurrently at once, while most CMOS sensors trigger the individual photodiodes sequentially, often described as a rolling shutter. Virtually all current smart device cameras use CMOS sensors with a rolling shutter [20]. Notably, researchers have used the rolling shutter to improve the sampling rate of OCC systems, both with displays [30] and blinking LEDs [31] as transmitters. Kuo et al. [32] propose using the rolling shutter in mobile phones to enable detecting LEDs blinking at different frequencies in their indoor positioning system achieving decimeter-level accuracy.

OCC research considers using several LED-based transmitters, such as indoor and outdoor lighting, displays, and flashes [20]. Approaches using displays as transmitters are referred to as screen-camera communication or camera-display communication [24]. In [33] Kurak et al. propose changing the colors of specific areas of a display, invisibly to the human eye, to transmit data at up to 16 bps. HiLight [34] suggests embedding data in the transparency, or alpha, channel of video or image frames so that the RGB channels do not require modifications. InFrame++ [35] uses differences between consequent frames. Furthermore, as discussed further in the next section, several proposed systems use barcodes displayed on screens and scanned with cameras to transmit information.

Screen-camera communication systems often suffer from framerate mismatch between the transmitter and the receiver. Common LCD displays use a refresh rate of 60 Hz, and response times of individual pixels in the displays varies greatly between display panel models. On the other hand, the frame rates of cameras vary between devices and depend on the environment and exposure requirements. Moreover, the programming APIs of smart devices may not provide exact control over the frame rate [36]. Generally, for efficient decoding, the exposure time should be significantly shorter than the periods of the transmitted signal. Some mechanisms have been proposed to address the frame mixing cause by the frame rate mismatch [37, 38].

Finally, some VLC research has accounted for the distance between the transmitter and the receiver. Miramirkhani et al. [13, 39] describe channel modelling for VLC of moving receivers or users based on the impulse response received at different locations relative to transmitting LEDs. In [39] the same authors leverage channel modelling of impulse responses at different distances in gas pipeline monitoring. The authors of [14, 15] propose a hierarchical coding scheme that essentially creates subchannels employing different levels of spatial detail. Such subchannels can transmit information over different distances: lower spatial frequencies require a lower resolution to properly resolve, thus increasing the receiving distance. The authors use the scheme to create an ITS system that sends information to vehicles approaching traffic lights through an array of LEDs. The system can transmit high-priority information to vehicles far away and then lower-priority data as the vehicles get close to the traffic light. The authors of [17] and [16] employ similar schemes to encode information at different levels of spatial detail into barcodes. These solutions, among others, are discussed in more detail next.

2.2 Barcodes

Barcodes are visual representations of data that, when combined with an optical receiver such a camera, can transmit information wirelessly over the visual channel. They have become widely popular thanks to their fast and accurate machine-readability [40, 41, 36]. Moreover, the cost of manufacturing barcode transmitters is extremely low: the cheapest transmitter is merely a print on paper, requiring no electronics. This is a great advantage in the most compelling use case of retail product markings, where the number of transmitters is significantly higher than the number of receivers. Other options, such as RFID tags, can sometimes provide better reliability due to the physical fragility of barcodes or unusual lighting scenarios, but impose a higher cost of manufacturing, suffer from RF interference, and tend to end up in waste [16].

The first generation of barcodes were one-dimensional (1D) sequences of bars and white space of specified widths. While dozens of different symbologies, or mappings of sequences to characters or digits, have been developed, two of them are the most popular options: UPC (Universal

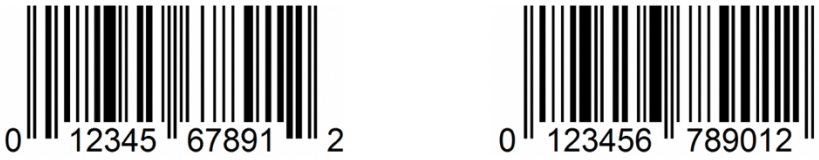

Figure 1. Examples of UPC-A (left) and EAN-13 (right) codes encoding the sequences of numbers visible in the codes.

Product Code) and EAN (European Article Number, officially International Article Number) codes. These codes, shown in Figure 1, are ubiquitously used for tracking trade items in retail stores, postal services, and warehousing. The data capacity of one-dimensional codes is limited. For example, the most commonly used specifications UPC-A and EAN-13 only encode 12 and 13 digits respectively, the EAN-13 actually being a superset of the UPC-A, augmented to include a country code or product type prefix. Several specifications with different properties and capacities for different purposes exist, such as the UPC variants A, B, C, D, E, 2, and 5, as well as the EAN variants 13, 8, 5, and 2. The low capacity is not an issue in the most common use cases of these codes, as the information encoded in the barcode is generally only an identifier that can be used to retrieve further information from a database.

However, the inevitable demand for higher capacities and the development of better scanner technology prompted the development of two-dimensional (2D), or bi-dimensional, barcodes. The first design approach of 2D codes, such as in PDF417 and Codablock F codes illustrated in Figure 2, was essentially a multiline stack of 1D codes [42]. For example, a Codablock F code is a stack of Code 128 codes in practice [43]. The stacked design provides considerably higher capacities with a single scan, but requires careful alignment of the scanner [41]. Thus, research quickly turned to matrix codes, such as the MaxiCode, Data Matrix, Aztec Code, and Quick Response (QR) codes depicted in Figure 3. The matrix designs have the distinct advantage of enabling efficient scanning even with a less precisely aligned scanner [41] while providing high data capacities: a QR code can hold up to 7089 characters.

Figure 2. Examples of PDF417 (left) and Codablock F (right) codes, both encoding the text "This is an example code".

Figure 3. Examples of MaxiCode, Data Matrix, Aztec code and QR code, all encoding the text "This is an example code".

Notably, the QR code, released by the Denso Corporation in 1994 [12] with the primary goal of easy detection and interpretability, has become widely used in mobile computing applications [41, 16, 44]. The structure of a QR code includes position detection patterns in three corners, typically the top corners and the bottom left corner, and a number of smaller alignment patterns distributed evenly over the code area towards the fourth corner. The number of alignment patterns depends on the code size: for small capacities it is often one. The detection and alignment patterns allow the code to be properly interpreted under distortion, such as the non-ideal perspective that is present in smart device scans as the camera sensor plane is virtually never perfectly parallel with the code plane. The maximum storage capacity of a QR code depends on the selected mode, or data type. The four possibilities are numeric, alphanumeric, binary, and Kanji characters. Furthermore, QR codes can trade off data capacity for reliability through error correction. For

example, a maximum data recovery level of approximately 30 percent can achieve a capacity of 3057 numeric characters, while a low 7 percent data recovery level can reach the maximum of 7089 numeric characters [36]. It must be noted that the higher capacities are achieved by adding more modules – i.e., blocks – to the code, in other words, by simply growing the size of the code. Overall, the properties of the QR code make it widely applicable and reliable, thus explaining its ubiquitous status.

The next step towards even higher capacities is the use of colors. The barcodes discussed above use only two colors, generally black and white, eventually allowing binary symbols only. This can be considered an on-off-keying (OOK) scheme where white blocks signify 0 and black blocks 1. Growing the constellation by using additional colors allows for one module to encode more information, consequently increasing the ratio of information per area when the module size is kept constant. This approach is called color-shift-keying (CSK): specific colors signify specific symbols. Usually, when encoding digital information, the number of colors follows 2^n where n is the number of bits encoded in one module. For example, 2 colors can encode 1 bit per module, 4 colors 2 bits, 8 colors 3 bits, etc. While selecting a scheme with a high n can theoretically lead to massive data densities, the probability of erroneous interpretation increases with the number of colors, requiring more redundancy for error correction and reducing the practical maximum capacity. The work in [40] shows a byte error rate of around 2 percent for a normal black and white QR code, while a similar four-color code has a byte error rate of 10 percent. They note that the error rates require respectively at least 4 and 20 percent of the code are to be used for error correction, leading to data densities of 96 and 80 percent, significantly reducing the capacity gain from using additional colors.

Examples of 2D color barcodes include Microsoft's High Capacity Color Barcode (HCCB) [45] and the HCC2D (High Capacity Colored Two Dimensional) proposed in [41]. The Microsoft HCCB was primarily developed to identify commercial audiovisual products such as movies and video games. It uses four or eight colors and an unusual triangular symbol shape. Microsoft notes data densities up to 3500 alphabetical characters per square inch using a 600dpi business card scanner. The HCC2D is essentially a multicolor QR code: it implements the same functional detectors and structure as a QR code while increasing the number of

colors, proposing similar robustness as a QR code but with a higher data density.

Barcodes have several challenges imposed by the visual channel. Several types of distortion and interference make the received information a mere estimation of the actual data transmitted. First, the transmitter may be partly occluded. In other words, parts of the barcode may not be visible to the scanner. This may happen, for example, due to physical damage to a printed barcode, glossy reflections on a screen displaying a barcode, or a third object blocking the line of sight. Some codes, such as the QR code, can overcome this with error correction methods. Second, there may be distortions caused by the receiver or its handling. Images may be blurred, the perspective may be distorted, and the location, orientation, and background of the barcode may vary. Third, the lighting conditions may pose issues, especially in mobile computing applications. There are several variables that can vary drastically, such as the ambient light, the screen brightness of the transmitter, and the exposure and white balance at the receiver. Next, in screen-camera communication, the moiré pattern [46] created by the display array aligning with the image sensor array can create significant noise in the colors received. Finally, the size of the barcode and the resolution of the receiver set a natural limit on the distance the barcode can properly be decoded from, as described by the Nyquist-Shannon sampling theorem [47].

Some of the challenges will simply restrict the sampling, while others mainly cause problems that can be tackled with clever code design and computer vision techniques [48, 49, 50].

2.2.1 Barcodes in screen-camera communication

Several approaches utilizing barcodes or barcode-like solutions for screen-camera communication have been proposed. While they largely focus on streaming data from one device to another by displaying a sequence of barcodes, several of the demonstrated design improvements are directly applicable to static barcodes, as they target challenges such as blurring and code location.

Hao et al. [51] propose COBRA (Color Barcode stReaming for smArtphones), a real-time data streaming VLC system designed for smartphones. They create a barcode optimized for small screens, cameras

with low framerates, and devices with limited processing power. The COBRA code uses a smart frame including color corner trackers and timing references that allow for fast, real-time interpretation even under blur and perspective distortion. The authors also introduce a blur-adaptation technique based on the degree of blur (DOB) metric. It leverages accelerometer readings at the transmitting device and adapts the code block size accordingly.

RDCode [52] aims at improving the throughput of screen-camera communication by focusing on reliability. It introduces a layered packet-frame-block structure, where blocks are groups of symbols or modules (that are traditionally considered as code blocks), frames are groups of blocks, and packets are sequences of frames. This layered structure allows for using different error correction schemes with different granularities. RDCode can double the communication rate achieved by COBRA.

LightSync [37] and Styrofoam [53] target inter-frame interference, or frame mixing, caused by the mismatch of framerates between the display and the camera. LightSync tries to interpret unsynchronized frames better instead of trying to match or separate the frames. Using per-line color tracking allowed by the rolling shutter property of CMOS sensors and inter-frame erasure coding to recover lost frames, they note doubled data rates compared to previous solutions. Instead, Styrofoam proposes inserting blank frames between code frames[1], which reduces the time the actual code is exposed but also reduces inter-symbol interference, leading to a total increase in data rate: the authors claim almost triple the performance of LightSync.

In [54], Wang et al. propose RainBar, an improved robust, high-data-rate VLC system. Compared to COBRA, RainBar removes the restriction of limiting the frame rate of the transmitter to half of the capture rate of the camera. In addition, the authors design an improved code layout featuring in-frame code locators for accurate code extraction and in-frame tracking bars for frame synchronization that works with high display rates.

[1] Styrofoam considers each code block a separate channel. In other words, frames are also block-specific. Thus, no fully blank codes are inserted, but the blanks are rather visible as white blocks in the code.

Some works also consider bidirectional communication, usually by utilizing the front-facing cameras of smart devices. CamTalk [55] implements secure bidirectional communication between smartphones using QR codes and a TCP-inspired session protocol. Given the inherent security properties of VLC, namely the ability to restrict communication to the area defined by line of sight, such bidirectional solutions have potential in security applications such as key exchange. Di Francesco and Montoya [36] propose a similar QR-based system for practical file exchange between smartphones.

Another branch of research has proposed encoding data in the frequency domain instead of the spatial domain [56]. Pixnet [17] implements a screen-camera communication system leveraging OFDM (orthogonal frequency-division multiplexing), more traditionally used in radio frequency technologies. The system suggests several potential improvements. First, perspective distortion, which is essentially irregular sampling of the code area, can be compensated for in the frequency domain by resampling the image after calculating the sampling frequency offset. Second, the effect of blur distortion increases with spatial frequency, or detail. Thus, the largely unaffected lower frequencies require less redundancy for error correction, and different levels of error correction can be used for different frequencies, requiring less redundancy on average. Finally, ambient light can be more easily ignored in the frequency domain, as it can be considered the DC frequency. Pixnet reaches data rates of 12 Mbps at a distance of 10 meters with consumer-grade LCD displays and digital cameras.

Focus [16] is another OFDM-based frequency domain approach, where the frequency domain is divided into subchannels formed by frequency ranges – levels of spatial detail. Figure 4 illustrates such a division. The authors construct the code by distributing data to these subchannels, noting that lower frequencies are more robust to decreasing sampling resolution, or increasing viewing distance. As described by the Nyquist-Shannon sampling theorem [47], given a sampling resolution, only frequencies below a certain limit frequency can be properly sampled. The limit frequency lowers as distance increases. Intuitively, as viewing distance grows, even larger spatial detail becomes blurred. Consequently, Focus is inherently a variable-rate code dependent on the channel quality at individual receivers, and it relaxes the trade-off between capacity and

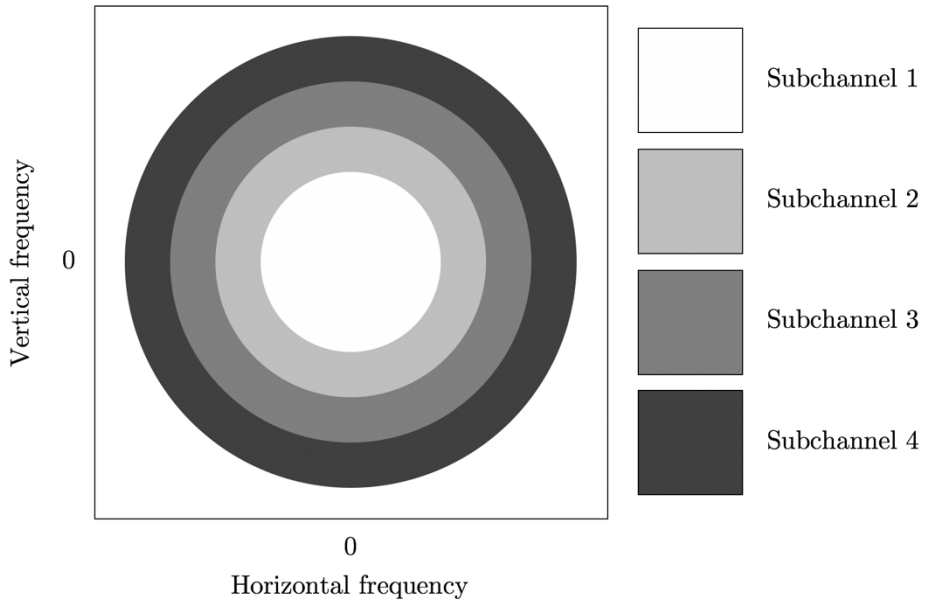

Figure 4. Illustration of subchannels based on frequency ranges.

reliability present in traditional barcodes with a single channel. The researchers suggest transmission rates up to three times those of Pixnet. In addition, Pixnet was designed with DSLR digital cameras, while Focus also targets smartphones.

These approaches based on spatial frequencies are the main existing barcode technologies that consider viewing distance, and thus the closest previous literature to the goal of this thesis. However, they primarily aim at improving the readability of the codes and extending the feasible scanning range by creating variable-rate codes. On the other hand, our goal is to explicitly define a separation zone between two groups of viewers. In essence, this means defining code modules, or blocks, that encode two different symbols based on the viewing distance.

3 System design and evaluation

The goal of the thesis is to define a barcode that can transmit different information to close and distant viewers. We explore two color-based methods that can theoretically achieve such distance-awareness: color-shifting of individual colors, and color-blending of color combinations.

Both methods target the following scenario. Assume that a barcode is to transmit two arbitrary sequences of bits to two users, close user A and far user B. Four distinct bit combinations are possible: 00, 01, 10, and 11, where the first digit indicates the bit for user A and the second digit the bit for user B. If both users should receive the same bit, no distance-dependency is required. However, if the users should receive different bits, a single module should present a different symbol depending on the distance. Figure 5 illustrates this generalized use case.

Our scope is limited to mobile computing applications. Consequently, all evaluation described below is conducted using smartphones as receivers

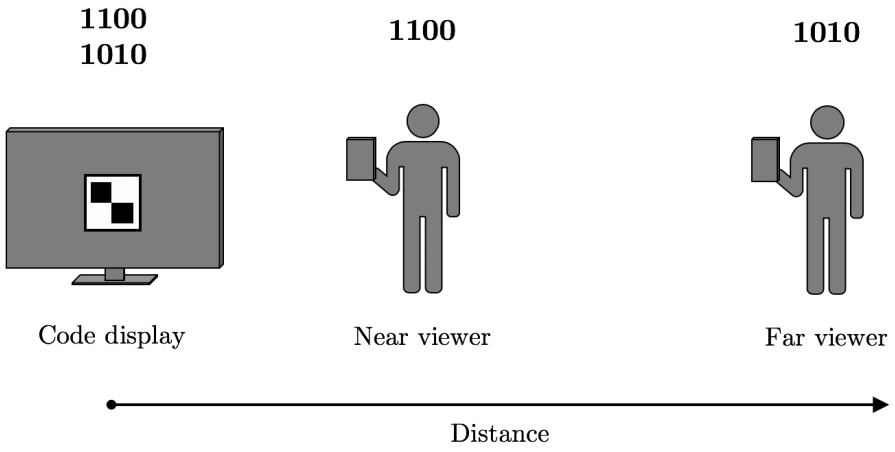

Figure 5. Generalized use case. The barcode displayed represents two separate bit sequences. When scanned, a near viewer receives the first sequence, and a far viewer receives the second sequence.

and common digital displays as transmitters. The specific devices are as follows. The receiver used is the primary back-facing camera of a OnePlus 6 smartphone with a CMOS sensor. The image resolution is locked to 10 megapixels, a commonly available smart device camera sensor resolution. Exposure and white balance are manually set constant to usual indoor office lighting unless otherwise mentioned. The transmitters used include the AMOLED display of a OnePlus 6 phone, the IPS display of an Apple MacBook Pro (13-inch, 2017), and the VA display of a large-format Samsung TV. All measurements are performed with the default color calibration settings of the devices and with maximum display brightness. In addition, we do not consider the effect of viewing angles. In other words, the camera used as a receiver is aligned parallel with the transmitting display directly in front of it and only the distance between the two is changed.

3.1 Color-shifting

3.1.1 Design

Color-shifting refers to the change in perception of a color at the receiver. For example, traditional LCD displays with a twisted neumatic (TN) display panel show a large amount of color-shifting based on the viewing angle, which has been utilized in some VLC applications [57, 58]. Notably, this depends on the display technology: modern in-plane switching (IPS) panels do not suffer from such narrow viewing angles. On the other hand, color shifting may happen due to viewing distance. Intuitively, the perception of a light source changes over distance due to signal attenuation or fading. The amount of attenuation depends on factors such as the transmission medium and the wavelength of the light. Therefore, light sources that combine different single-wavelength sources, or primaries, to produce a combination source will experience shifting in the ratios of the primaries. Indeed, LCD displays utilize a subpixel array of different primary colors to allow a single pixel to produce several colors. As a consequence, they will show some shifting in the ratios of the primaries over distance, theoretically leading to color shifting based on the viewing distance in the aggregate color.

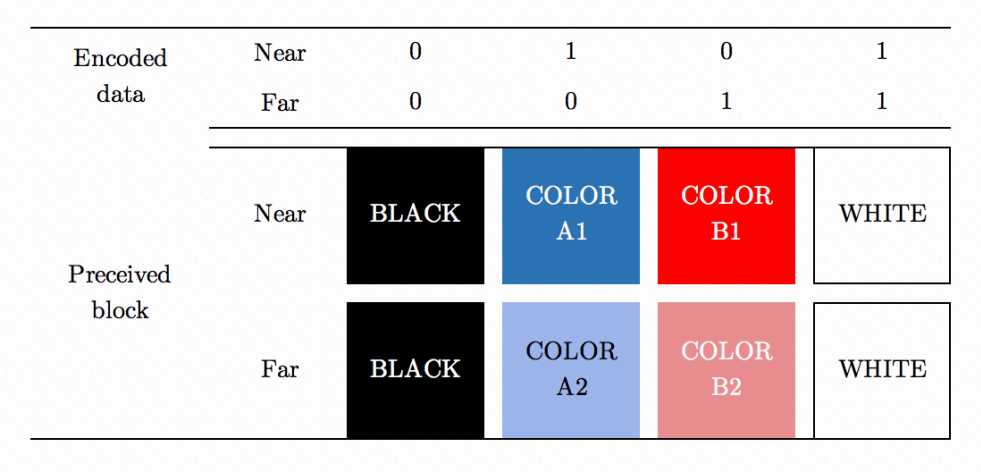

Figure 6. Color-shifting modules representing all possible 1-bit combinations. Color A2 is a shifted, or faded, version of color A1, and color B2 is a shifted version of color B1. Blue and red coloring is only used for illustration and is not representative of actual color shifting.

Now, assume the scenario described above, where a barcode transmits two different bit sequences: one to a close user A and another to a far user B. Using color shifting, we define the system illustrated in Figure 6. In the module system, bit combinations 00 and 11 can be represented with a single color for both users, since both receive the same value. We define 00 as black and 11 as white. For the combination 10, color A1 is shown at the transmitter. User A (the close viewer) receives approximately the same color, while user B (the distant viewer) receives an attenuated version of the signal, color A2. The same is applied for the bit combination 01 with a different color and inverted bits. Consequently, the combination of any two arbitrary bit sequences can be represented as a sequence of modules or barcode blocks.

Alternatively, the following two-step system could be defined. First, the receiver determines the distance between the transmitter and itself based on the colors received over the whole barcode. Second, it decides on a constellation to use based on the distance estimation, and each block is decoded based on the selected constellation. In other words, after the selection of the constellation, the color-shifting of individual blocks is no longer considered. For example, the constellation can define the color red as 0 or 1, and the very specific tone of red is inconsequential as long as it

can be clearly separated from the other colors used. The benefit of this approach is that more than one color can be used to deduce the distance, theoretically leading to more robust estimation than when the interpretation is based on only one block[2]. On the other hand, this requires more computation for the explicit distance estimation.

The following section presents measurements on color-shifting in screen-camera communication scenarios with smart devices and evaluates the feasibility of color-shifting-based approaches. We perform our own empirical measurements as the previous works leveraging color-shifting properties of digital displays consider TN panels whereas we mainly consider modern smart devices with AMOLED and IPS panels.

3.1.2 Evaluation

In order to capture a basic channel model, we define the following simple measurement. A grid of nine colors is displayed on an IPS screen in simple barcode form, and a smartphone camera is used to take pictures of the grid from several distances, measuring the received RGB values. Some sensor noise is expected. Thus, we use an average of the neighboring pixels for the center pixel of each color. In addition, the image is slightly blurred with a Gaussian filter to reduce noise from the moiré effect before reading the values.

The first round of measurements is taken with automatic exposure and focus based on the center point of the image, targeted at the color in the center of the grid. As expected, automatic exposure varies enough to be clearly visible in the received RGB values as large average intensity changes. In other words, all three RGB channels increase or decrease simultaneously.

Consequently, we set the exposure manually to support average office lighting conditions and minimize external noise. The results are presented in Figure 7. Some drifting in the individual channels is visible, pointing to the existence of color shifting. Note that the large reduction in average intensity at around 50 centimeters is largely due to the maximum of the moiré-producing aliasing with the used display-sensor combination

[2] Excluding the worst-case scenario where both encoded bit sequences are identical and all bits are the same (i.e. 111... or 000...).

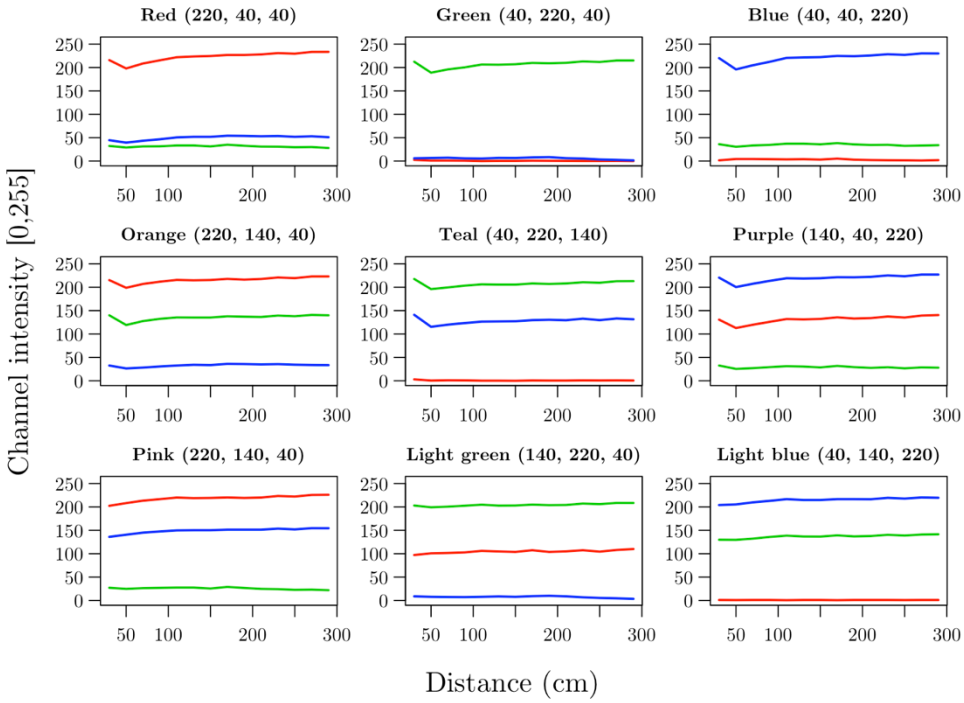

Figure 7. Color-shifting measurements for distances from 30 cm to 300 cm with 20 cm increments. Colors are displayed on an IPS display and received on a smartphone camera with a CMOS sensor.

matching that distance. This creates a clearly visible dark pattern over the image, thus significantly reducing the average intensity.

We then define a simple channel model with multiple linear regression as follows:

$$distance_i = \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_{27} x_{i27} + \varepsilon_i,$$

where x_1 through x_{27} are the 27 individual RGB channels for all nine colors. Next, we use this to predict the distance an image is taken at. Results for a test set of images taken in the same conditions as the training set are presented in Figure 8. The distance predicted based on the received colors follows closely the actual distance, the maximum absolute error being only 27.28 centimeters in this test set. Moreover, the predictions are almost monotonically increasing with the 20 cm step size, excluding two observations. These two exceptions are likely due to small uncontrollable

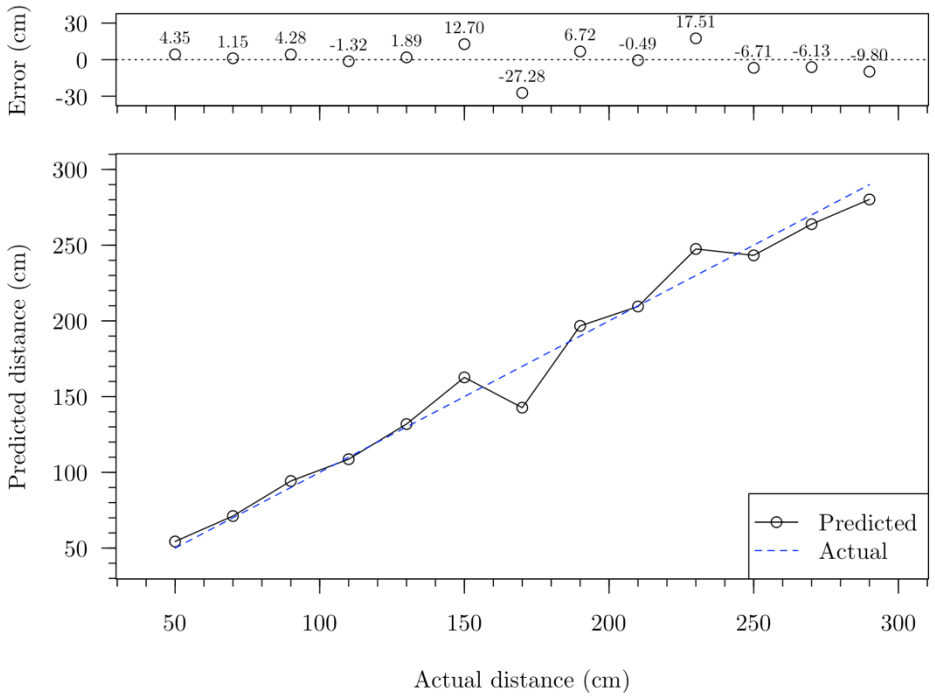

Figure 8. Distance predicted on received colors in ideal conditions.

changes in ambient lighting as well as sensor noise. Therefore, it would be feasible to define limits for close and far viewing distances in order to utilize this scheme in a barcode.

However, test sets captured in non-ideal, variable conditions lead to a very different conclusion. For the scheme to be practically feasible, it needs to be robust to expected, unavoidable variation in external conditions. While internal factors such as exposure and white balance can be locked in software, external factors such as ambient lighting and display brightness cannot be controlled to the same level. We repeat the distance predictions in several indoor conditions, such as office rooms, classrooms, auditoriums, and lounge spaces to evaluate the sensitivity of the system. We note that, while the predicted distance overall tends to increase with the actual distance, the actual numbers do not correspond to the physical environment. Thus, setting limits for close and far viewing distances becomes non-trivial if at all possible. For instance, Figure 9 presents distances predicted based on images taken in another office room. Now, the prediction errors are in the magnitude of meters, growing to

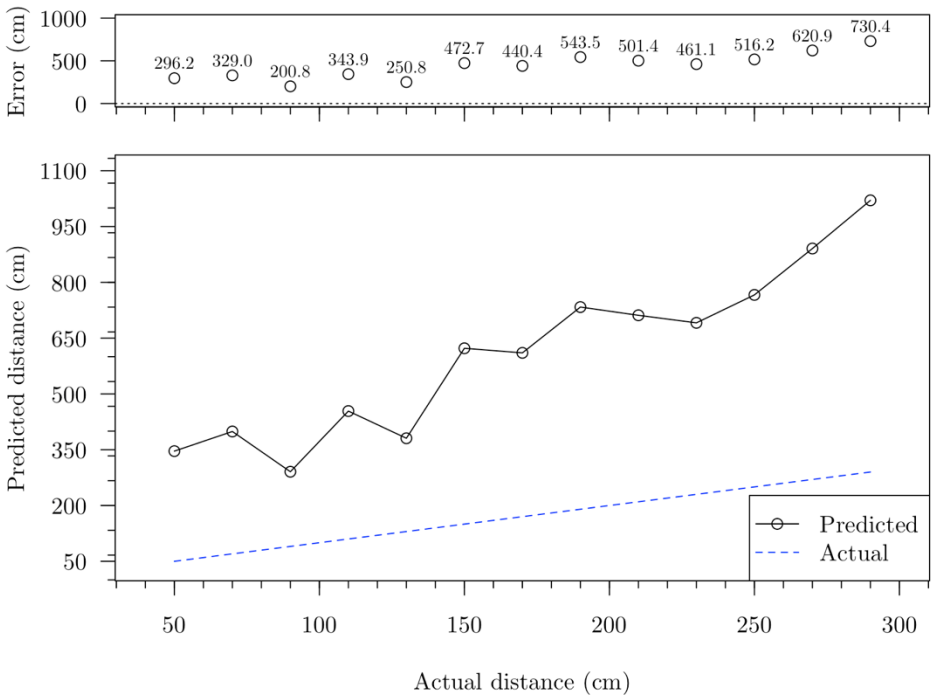

Figure 9. Distance predicted on received colors in non-ideal conditions.

over seven meters at the actual distance of three meters. Crucially, the value the model predicts at the closest actual distance (3.5 meters) is greater than the furthest actual distance included in the measurements (3 meters).

This is intuitively supported by the small magnitude of the color shifting measured. Table 1 shows the standard deviations and maximal differences for the measurements in Figure 7. Given that the unit step defined by the RGB color model is one, the deviations of a few units are easily overrun by external factors including ambient light, display brightness, and display calibration. Notably, consumer devices generally aim for pleasant and bright colors instead of accurate color reproduction. Such a goal is highly subjective, leading to highly manufacturer-dependent color reproduction bias. Furthermore, several devices have features that purposefully distort the colors for purposes such as night modes in smartphones and computers, and content-specific color modes in TVs [59].

In conclusion, the empirical measurements demonstrate that color shifting exists and can be utilized in ideal conditions. However, any non-

	Red			Green			Blue		
	R	G	B	R	G	B	R	G	B
Stdev	10.08	1.90	4.56	0.66	7.48	1.96	1.28	2.06	9.88
Max-Min	35.50	7.00	14.80	2.60	26.00	6.90	4.00	7.80	34.20
	Orange			Teal			Purple		
	R	G	B	R	G	B	R	G	B
Stdev	6.39	5.79	2.86	0.72	5.51	6.15	7.52	2.03	7.43
Max-Min	24.20	21.60	9.80	3.00	21.60	25.60	27.70	7.20	26.60
	Pink			Light green			Light blue		
	R	G	B	R	G	B	R	G	B
Stdev	6.48	2.07	5.33	3.30	2.73	1.82	0.15	3.82	5.01
Max-Min	23.60	7.00	18.60	12.80	9.40	6.50	0.40	11.90	16.20

Table 1. Color-shifting measurement deviations for the measurements shown in Figure 7.

ideal conditions render it unusable for practical scenarios involving common smart devices and mobile computing applications. Channel calibration [60] may solve this issue for certain applications, but it is not directly applicable to single-observation use cases such as barcodes.

3.2 Color-blending

As a result of the lack of robustness of the color-shifting method discussed above, we consider a different method of producing distance-dependent code modules. Such a method is based on color blending, or color dithering and is discussed next.

3.2.1 Design

Color blending occurs when two colors overlap either in the transmitter or the receiver, forming a perception of a third color. This is strongly related to dithering, i.e., the intentional application of noise, which has been widely applied in digital imaging to alter the bit-depth of a source. For example, the color reproduction of LCD displays can be improved by leveraging dithering algorithms to enable 8-bit displays to seemingly produce 10-bit output without actual 10-bit hardware, moving from a discrete color set closer to continuous color representation [61, 62, 63]. On

the other hand, bit-depth can be lowered while keeping a graceful representation of the source, thus allowing data compression [64].

While these methods generally operate on individual pixels, the color-blending method for distance-sensitive barcode modules uses this notation on a larger scale. The key observation is that any perceived blending between modules requires a low-enough receiver resolution. In other words, given a fixed receiver resolution, color blending between fixed-size barcode blocks increases with distance. This is intuitively analogous to blur distortion. The basis of this method is partially related to frequency-based approaches such as [15] and [16], in that the distance-dependent capability of a receiver to resolve spatial detail is leveraged to create different channels. However, code generation is considerably different, as we do not directly encode information in spatial detail using a transform.

To leverage blending, we define a similar module scheme as in the color-shifting method discussed above, but replace color-shifting with color-blending. Again, consider user A at a close viewing distance, and user B at a far viewing distance. When both users should receive the same bit, only a single module is required, as there is no distance-dependency. On the other hand, when the users should receive different bits from the same module, we require a distance-sensitive module. We define this module as a macroblock that consists of four microblocks, or sub-blocks, in two predefined colors alternating such that the center of the macroblock will experience color blending when viewing distance grows. The scheme is illustrated in Figure 10. For 1-bit modules, we require two such modules. In the sample implementation, we define these as the red-blue module and the red-green module. According to the additive RGB color model and the hue circle, these produce purple and yellow when blended, respectively.

Now, given the definition of the modules, the receiver can decide on the bit based on the perceived color hue histogram of the block. For example, a close observer of the red-blue block will see two clearly separable peaks around blue and red, while a distant observer will see the peaks blend into each other around the purple hue, as depicted in Figure 11.

In practice, this can be efficiently implemented by quantizing the image of the block to a number of hues and defining a set of thresholds for the proportions of hues observed in the image. For the 1-bit case with

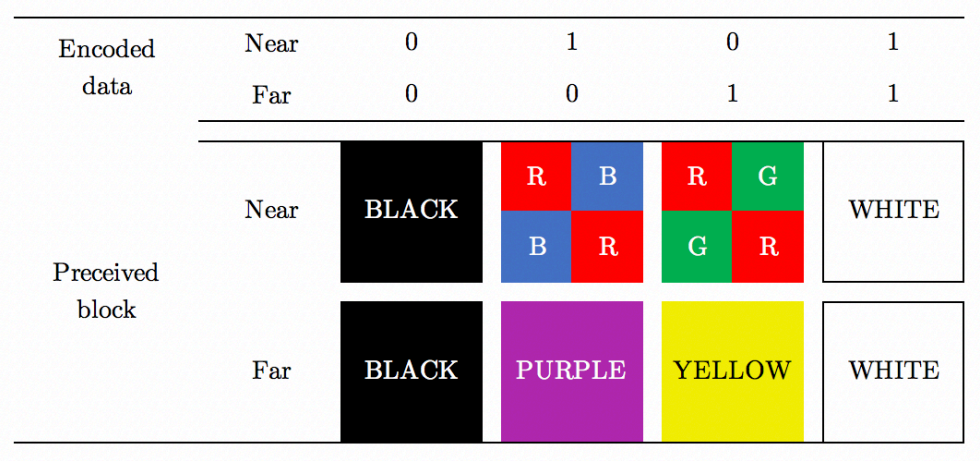

Figure 10. Color-blending modules representing all possible 1-bit combinations. Blending occurs according to the additive RGB color model.

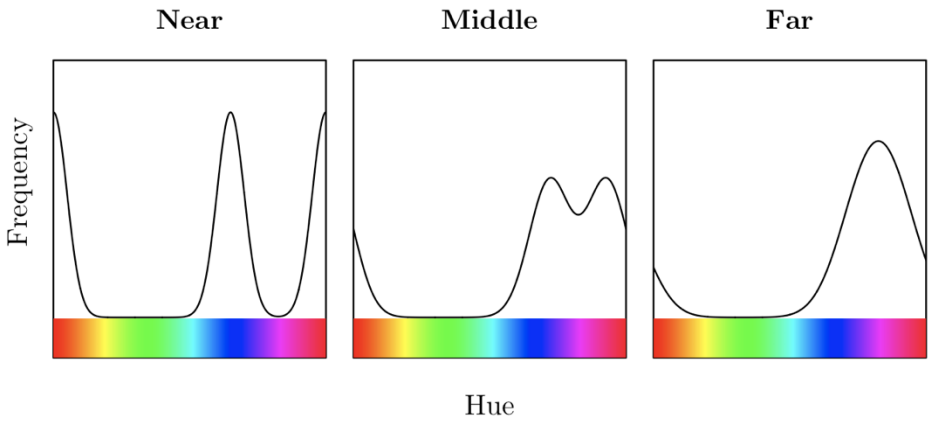

Figure 11. Illustration of expected color blending over distance. The histograms represent the received color hues at near, middle, and far viewing distances for a macroblock of red and blue microblocks, as described in Figure 10. As distance grows, blue and red blend and are increasingly perceived as purple. Note that the x-axis is periodic since it represents the hue circle.

the colors discussed above, the 60° six-way division of the hue circle into red, yellow, green, cyan, blue, and magenta is a natural choice of quantization. However, the thresholds that define the decision rule set clearly depend on the desired separation properties: the closer the desired separation distance, the lesser blending is required. A specific sample implementation of the rule set is described in Chapter 4 along with the sample application implementation. Moreover, note that while the color-shifting discussed above is naturally also expected to occur here, the magnitude of the shifting is overrun by the quantization given high-enough hue separation.

Finally, this scheme suffers from the capacity-robustness trade-off in terms of potential capacity as compared to the color-shifting scheme. A single module requires three colors instead of one: the two primaries and the blend, or in terms of the hue circle, two points on the circle and the whole curve between them. However, this is not necessarily a problem, as the target use cases only require enough capacity for short URLs or small tokens. In other words, the code is used as a link to another larger resource.

3.2.2 Evaluation

To evaluate the color blending scheme, we construct a simple barcode as a 20-by-20 grid of modules. This provides 400 bits or 50 bytes of total capacity in binary data. The structure includes a 5-byte header (1 byte for data size and 4 bytes for a CRC32 checksum) and the actual payload. Thus, the maximum payload size with a 20-by-20 grid of code modules is 45 bytes; this is enough for relatively short URLs or other token-like content. An example barcode is shown in Figure 12.

The scheme shows clear capability to produce distance-dependent output. In this regard, Figure 13 and Figure 14 show the distance separation as success counts out of six trials per distance for two different code sizes. Black indicates cases where the output of the scan was the data targeted at a near viewer and checksum verification succeeded, stripes where checksum verification failed, and white where the data targeted at a distant viewer was successfully read. Ideally, at least a single striped bar should separate mostly black and mostly white bars as indication of a physical separation zone.

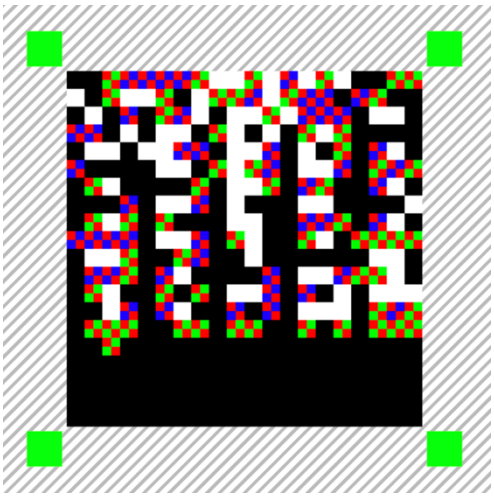

Figure 12. An example barcode based on the color blending scheme. The striped background is not part of the code design.

Figure 13 presents the results for a 13-by-13-centimeter barcode displayed on a laptop with an IPS screen. Given the structure of the code, a single macroblock is 0.52 cm in width and height. From a viewing distance below 80 cm, the decoding begins to fail and is only successful again at around 120-140 cm, producing a separation zone of approximately half a meter. Results for a smaller code at 6.5 by 6.5 cm, displayed on a smartphone with an AMOLED screen, are shown in Figure 14. In this case, an individual macroblock is sized at 0.26 cm. Consequently, the separation zone is closer and narrower than with a larger code size: from below 40 cm to around 50 cm. The distant (right) end of the charts is inconsequential: it merely represents the gradual degrading of the channel quality such that the resolution is no longer high enough for successful decoding or even for locating the code from the background. Notably, the code constructed for this evaluation is not designed for robust location but it is only meant as a demonstration of the distance-aware modules, as this is the scope of the thesis. Clearly, when the blocks defined in this scheme are embedded in a more robust structure, such as that in QR codes, the far end will reach further.

Figure 15 presents examples of code blocks received at different distances for different transmitting devices to show color blending in practice. The images shown are from the late processing steps of the

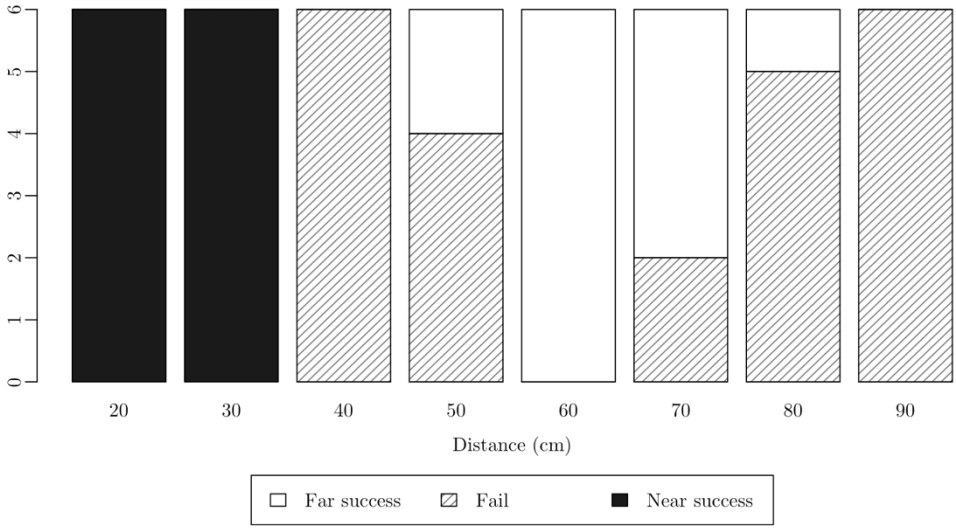

Figure 13. Code output by viewing distance with a 13 cm code displayed on an IPS laptop screen. Black bars indicate cases where the near data was received, striped bars cases where checksum verification failed, and white bars cases where the far data was successfully received. The fully striped area between black and white blocks indicates the separation area.

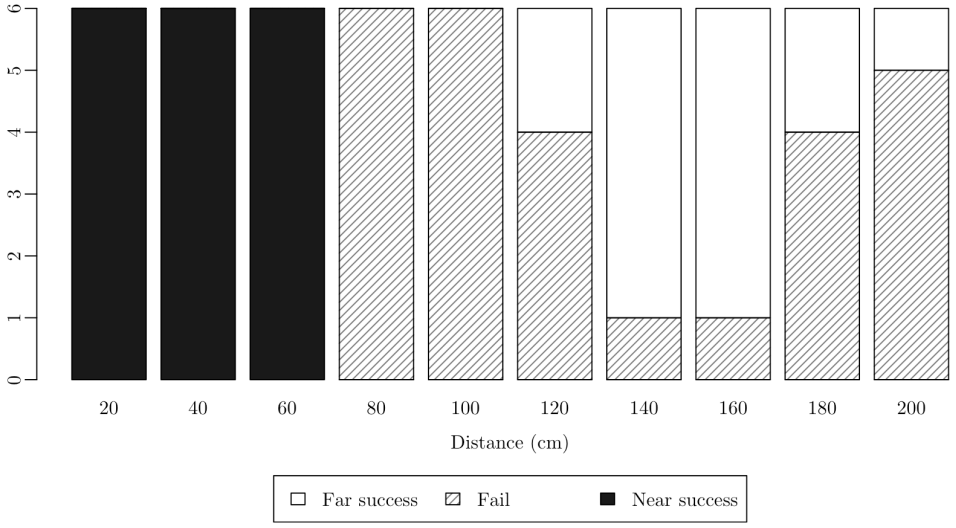

Figure 14. Code output by viewing distance with a 6.5 cm code displayed on an AMOLED smartphone screen. See Figure 13 above for further explanation.

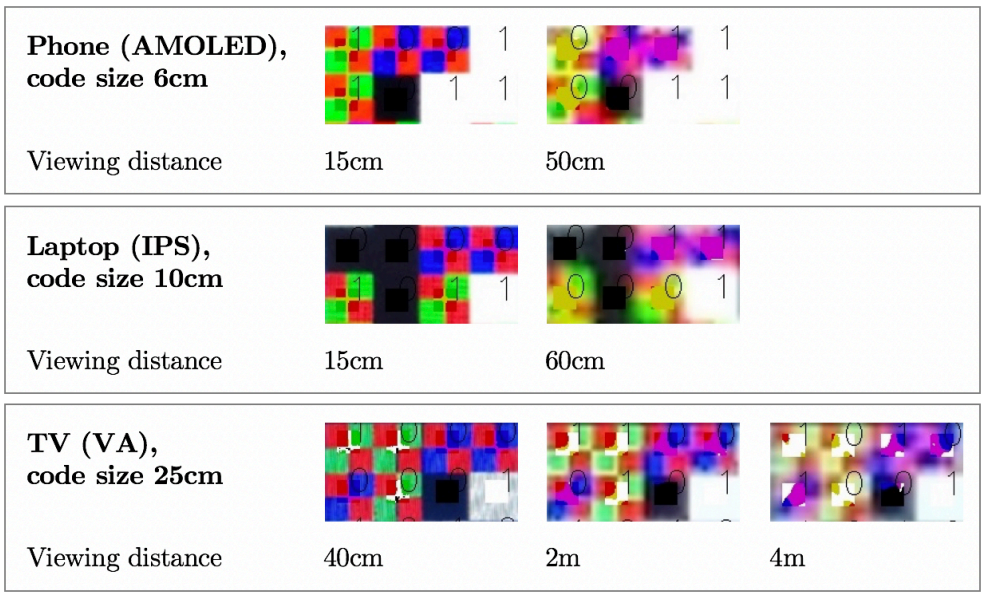

Figure 15. Examples of received code blocks for different transmitters and distances. An area of 21 by 21 pixels in the center of the block is quantized and the output bit is decided on based on the quantized area: the numbers over the code blocks show the perceived bit. The display panel type is mentioned in parentheses.

application and show both the quantization at the centers of the blocks and the decoded bits for individual blocks. In the images taken close to the transmitter (the first column), there is little blending between the microblocks, independent from the screen type. Some blending always occurs when pixels along the edges do not perfectly align between the transmitter and the receiver, when pixel densities are different, and due to compression in image formats. This is seen, for example, as the thin yellow and purple lines in the top left image. On the other hand, the images taken from a greater distance show a large amount of blending within the quantized center areas. In the example cases on AMOLED and IPS displays, the blending occurs virtually over the whole center area of 21 by 21 pixels in the processed image, and very little of the microblock colors can be seen in the corners. Therefore, bit decisions are seemingly trivial.

However, the example images of codes displayed on a Samsung TV with a vertical alignment (VA) panel show that decoding is sensitive to

the display type. Specifically, the images of a VA panel exhibit a higher amount of white bleeding near green areas, as can be seen in all images on the third row of Figure 15. Consequently, the rule set that was designed over AMOLED and IPS displays would need to be expanded to accommodate for this white bleeding. This can be seen as some incorrect interpretations of bits in the third row of images in Figure 15. Nevertheless, the quantized areas show an increasing amount of color blending between the primary colors of the microblocks as viewing distance grows. In other words, while the decision-making method needs expansion, the fundamental phenomenon is still present. Alternatively, instead of adapting the decision-making, the colors could be changed to account for the perceived increasing intensity or luminance in the green channel, either by avoiding the green-cyan part of the spectrum or by reducing the luminance of the displayed green blocks. However, any such solution comes with a trade-off in either capacity or robustness in the quantization.

Furthermore, evaluation in different environments shows that, as long as the exposure remains the same, the blending patterns experienced remain virtually identical within the same panel type in several common indoor scenarios. This is intuitively explained by the robustness of hue quantization: small changes of several intensity steps in the RGB channels caused by external factors, such as ambient lighting, do not change the quantization result as the hues have a wide 60° separation. Moreover, intensity changes that are constant across all three RGB channels do not change the hue of a color. Thus, small exposure changes are largely negligible. However, since hue quantization needs to be augmented with intensity thresholds to recognize black and white, large exposure changes will eventually turn some colors into black or white. Nevertheless, the color resolution required by the color blending scheme is inherently lower than that required by the color-shifting method, leading to a considerably more robust system.

In conclusion, the empirical evaluation demonstrates that color blending is effective and robust in practical scenarios with common hardware. Thus, it can be leveraged for distance-aware 2D barcodes. However, the experiments reveal certain differences in the blending over different display types. Specifically, the amount of perceived high intensity or white bleeding between microblocks is higher on VA panels

commonly used in large-format TV screens. Accordingly, the decision-making process needs to be adapted to guarantee wide applicability. There are several ways of doing this. Most obviously, the thresholds for color proportions in the hue histogram method can be changed to accommodate for this. Alternatively, given the distinct visual changes in the appearance of code blocks, including the shape of the blending, elementary machine learning methods will be able to more robustly classify code modules at the cost of slightly increased interpretation time or processing power and power usage requirements.

4 Proof-of-concept application and use case

This chapter further details the practical implementation of our system. First, we describe the mobile application developed for the evaluation presented in Chapter 3, including the specific rule set we use for bit decisions with the color blending modules. We use OnePlus 6 smartphones; thus, the application is developed for Android. Second, we present our sample use case implementation: a presentation system where the presenter and the audience can use the same barcode to access different tools.

4.1 Android application

4.1.1 Code generation

Code generation is simply as follows. First, the header (1 byte for data size and 4 bytes for a CRC32 checksum) is prepended to the payload. Second, the resulting byte sequence is split into individual bits. This is done twice, as there are two separate payloads, one for the near viewers and one for the far viewers. Third, code modules are decided based on the bit combinations of the two sequences, using separate modules for combinations 00, 01, 10, and 11, as described above. Finally, the code modules are drawn on the grid and squares with a specific green color are added to the corners of the code for code location in images.

4.1.2 Code scanning

The code scanner is implemented with the Android Camera2 API and the OpenCV SDK 4.0.0 compiled with OpenCL support.

First, a 10-megapixel image is taken with manual settings: aperture at f/1.7, shutter speed at 1/30s, ISO at 80, and white balance set to daylight. These settings are suitable for imaging LCD displays in indoor scenarios in situations where the display brightness is not extremely low. The

sensitivity is set to the lowest value supported by the camera module to minimize errors due to internal, sensor-produced noise[3].

Second, the code is located in the image based on the four green corner locators by thresholding the image for the specific green color, slightly eroding the threshold image, searching for contours of a sufficiently large area, and choosing four contours as the corners based on minimal and maximal vertical and horizontal coordinates.

Third, a perspective transform is performed, producing a fixed-size square image, and the code is smoothed with Gaussian blur to reduce moiré and sensor noise in preparation for code module extraction. Note that a Gaussian filter will also introduce some non-channel-induced color blending, and other smoothing methods such as bilateral filtering would better retain edges between different colors. Nevertheless, the choice of the smoothing method is inconsequential as long as it is acknowledged in the process of making the bit decisions. Thus, implementation simplicity and speed are preferred.

Next, each code module, or block, is read as the area defined by the midpoint and its neighbors within ten pixels horizontally and vertically, leading to a 21-by-21-pixel area representing the module. For the color-shifting module defined in Section 3.1, the output is simply an average of the values in this area. On the other hand, for the color blending scheme described in Section 3.2, each pixel is quantized to a hue in the HSV color space, or to black or white if the luminosity is sufficiently low or high. The output is then the quantized module area, conceptually a histogram of the color frequencies of this area. The following only considers the color blending scheme, as color-shifting was empirically found too weak for practical uses, as described above.

The module is interpreted as a 0 or 1 bit based on the color frequencies in the 441 pixels. First, we define a set of proportions: ubiquity as 98%, dominance as 65%, minority as 35%, and existence as 2%. Then, the following logic, in order, is used to decide on the bit. If black exists, the block is considered black. If white is ubiquitous, the block is white. If yellow is dominant, the module is yellow, and if purple is dominant, the module is purple. If green exists and red and green combined are over the

[3] The lowest supported ISO value will likely differ between specific models or manufacturers.

minority threshold, the module is interpreted as a red-green module. If blue exists and red and blue in total are over the minority threshold, the block is considered a red-blue module. If yellow exists and the frequency of yellow is higher than the combined frequency of red and green, the block is yellow. If purple exists and is more frequent than red and blue in total, the block is purple. Finally, if these fail to recognize the module, the bit is decided based on the highest frequency, the highest peak of the histogram.

The rule set described above accounts for several properties of the channel experienced during evaluation. For example, low luminance tends to get suppressed by high luminance. In other words, bright areas bleed to dark areas, leading to black modules getting slightly suppressed by any other block. Consequently, a block needs to be interpreted as black at any existence of quantized black. Similarly, yellow and purple blocks that tend to get dominated by white but are not actually white modules need to be accounted for. This rule set succeeds in properly producing a separation zone between near and far users in all indoor scenarios tested using IPS or AMOLED displays. However, given the additional white bleeding occurring with VA panels discussed above, as illustrated in Figure 15, another conditional structure should be added to properly support VA panels.

Finally, the bits are combined into bytes, data integrity is verified with the checksum, and the decoded payload is output if the verification succeeds. Note that the separation zone we observe is a direct consequence of the checksum verification; blocks of the same type get interpreted differently close to the threshold limits due to subtle differences caused by noise.

4.2 Sample use case

There are several potential use cases for distance-aware visual light communication and barcodes, such as hierarchical labeling in warehousing a logistics, signage in indoor navigation and positioning, intelligent traffic systems, and presentation systems. Generally, distance-awareness can provide either hierarchical, additive information based on channel quality, or separation between users of different roles. We showcase a use case in

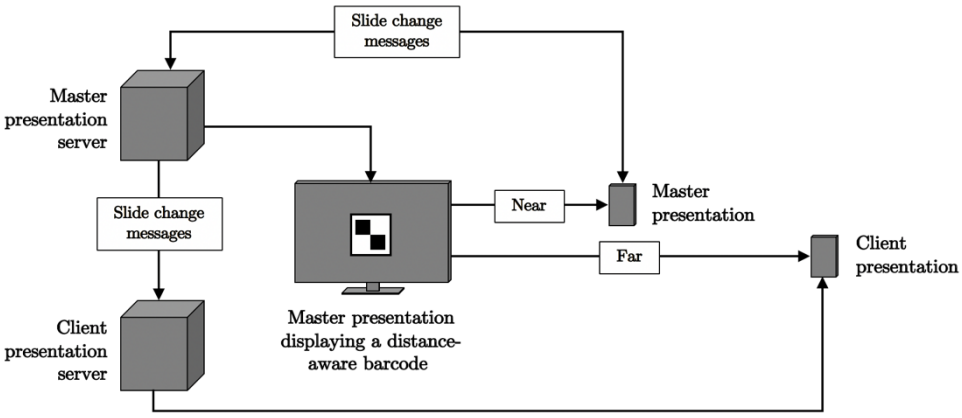

Figure 16. Presentation system use case structure.

presentation systems where the near and far users are defined as the presenter and the audience, similar to [65].

The goal of the use case is to embed into the presentation a code that provides the presenter quick access to presenter tools such as speaker notes, and the audience quick access to auxiliary material to follow the presentation. Our implementation defines two versions of the presentation: a master that controls all presentations including other masters, and a client that follows the master without affecting any other presentation. In other words, the master can be used by the presenter as a remote control for slide changes for all presentations, while the client can be browsed by the audience without affecting slides on any other presentation. The structure is depicted in Figure 16.

We implement the showcase with RevealJS [66], an HTML presentation framework that provides support for the master-client structure. One web server provides the master presentation, another serves the client presentation, and these are connected by a socket.io [67] server that delivers slide synchronization messages. All servers run on a single Amazon EC2 micro instance and are served through separate ports.

We embed a custom code generation plugin into the master presentation to show a color blending code on the first slide which is typically presented at the very beginning of a presentation. The plugin is simply a JavaScript port of the code generation described in Section 4.1. It dynamically catches the server IP and ports for the master and client

presentations, and draws a color blending distance-aware code accordingly to an HTML canvas that can be embedded into the presentation. Consequently, when scanned with the Android application, the code outputs links to either the master or the client presentation based on the viewing distance, and the links can be opened with any browser available on the smartphone.

Several presentation frameworks provide multiple presenter-aid features such as remote controls, stopwatches, speaker notes, and polls. While we choose to demonstrate our code with the simple remote-controlled master-client slide synchronization example, implementing another feature, such as speaker notes on the presenter's phone instead of a copy of the master presentation, is as simple as enabling the feature and changing the URLs encoded in the barcode.

Furthermore, in some cases it may be desirable to only include presenter features with no audience features. Notably, this can be implemented with a traditional barcode. However, a distance-aware barcode can provide a low-security physical barrier that restricts use over a certain distance. This can be achieved by filling the far channel with random data, or by simply leaving it empty. With the evaluation code we implement, it is enough to ensure that the checksum verification always fails for the far channel. Naturally, this is theoretically equal to reducing a traditional barcode to a size that cannot be decoded by further-away viewers, but practical implementations may benefit from not having to change code sizes.

5 Discussion

The distance-aware color blending scheme discussed above merely defines a code module system. Therefore, it can be embedded in any block-based 2D barcode structure, such as the QR code, to augment it with distance-sensitivity. Maximal viewing distance and capacity are comparable to the selected host structure when a traditional binary (black or white) block is directly replaced with a macroblock in the color blending scheme. Usually, close viewers have ample resolution to decode considerably smaller spatial detail than required for a simple black-or-white binary block, allowing for the direct replacement. For cases with low resolutions, the block size can be increased, or binary blocks can be replaced with microblocks instead of macroblocks. Since the module scheme we propose uses a minimum of four microblocks in one macroblock, a direct replacement of binary blocks with microblocks leads to a capacity one quarter that of the host, at highest.

In terms of potential data capacity, the color blending system places between traditional binary blocks and full color-shift-keying systems using several colors. For a one-bit scheme, two actual color-blending modules are required, allowing large hue differences. This provides high robustness in color quantization, thus allowing for the wide practical applicability observed in our empirical evaluation. For a two-bit scheme, already 12 color combinations are required for the differing bit combinations, considerably reducing the robustness to noise and thus wide usefulness in practical applications if very high capacities are required. However, non-binary data representations can utilize schemes between the one-bit and two-bit capacities.

For practical applications, capacity and distance-sensitivity are largely independent. While capacity and separation zone are both affected by module size, bit decision rules can be defined such that the separation zone is always low enough in practical use cases. For example, although there is a clear difference between the location and the width of the separation zone in the evaluation with different block sizes discussed in Section 3.2, the separation distance remains sufficiently low in all scenarios. The exact distance is inconsequential in many use cases. For

instance, a presenter and an audience are commonly separated by a distance that is considerably larger than the separation distance from the transmitter. Thus, block size can be somewhat freely defined to accommodate capacity and code reach requirements. Moreover, the module system can be adapted for very large sizes, such as auditorium presentation use cases, by simply increasing the number of microblocks inside a macroblock. For example, four columns and rows can be used, decreasing the size of microblocks and therefore decreasing the separation distance. As long as the area inside the macroblocks is equally allocated to the two primary colors, identical hue histogram behavior is retained.

6 Conclusion

In this thesis, we implement a distance-aware 2D barcode for mobile computing applications. We propose and evaluate two color-based methods of achieving the distance-sensitivity: color-shifting and color blending. Color-shifting relies on the perceived changes in the color tone over distance. Our empirical evaluation shows that a modern smartphone camera is capable of receiving colors accurately enough to observe consistent color-shifting in ideal conditions. However, small changes in the transmitting display or external conditions, such as ambient light, render the magnitude of the color-shifting effect too small for any robustness in practical use cases. On the other hand, the color blending approach leverages predictable blending of two primary colors over distance due to decreasing effective resolution. Evaluation shows that the blending is not only observed as expected but also considerably more robust to non-ideal conditions than color-shifting, thereby allowing for practical use in most indoor scenarios.

Our implementation captures the level of color blending based on the perceived color histogram over a code block. It achieves our goal of creating a clear separation zone between a near and a far viewer. The separation distance depends on the code module size: a block size of 0.52 cm sets the center of the separation zone at around 1 meter, while a block size of 0.26 cm sets the separation at approximately 45 centimeters. Similarly, the width of the separation zone grows with module size. We note that while our implementation is robust on AMOLED and IPS displays, it must be augmented to better support VA display panels used in some large TVs. However, this is merely an implementation issue, as the blending is clearly observable in all panel types tested.

The distance-aware system we propose has several potential applications, such as labeling in logistics and warehousing, indoor navigation, and presentation systems. We describe a sample implementation of a presentation system giving the presenter quick access to a master presentation and the audience an auxiliary client presentation to browse independently from the master.

Overall, the color blending module system shows the feasibility of color-based, distance-aware 2D barcodes for mobile computing applications with no special-purpose hardware.

Finally, there are two main opportunities for further research. First, the color blending module system can be embedded in a code structure designed for robust location and interpretation, such as the QR code frame. Second, the bit decision making method can be improved to support more display types, including projectors and paper, and to provide more accurate bit decisions. Moreover, alternative methods, such as machine learning, can be employed.

7 References

[1] We Are Social, "Global Digital Report 2018," 2018. [Online]. Available: https://digitalreport.wearesocial.com/.

[2] Ericsson, "Ericsson Mobility Report," June 2018. [Online]. Available: https://www.ericsson.com/assets/local/mobility-report/documents/2018/ericsson-mobility-report-june-2018.pdf.

[3] Cisco, "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update," February 2017. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.pdf.

[4] Wi-Fi Alliance, "FCC must proceed on 6 GHz NPRM to meet growing spectrum demand," 13 September 2018. [Online]. Available: https://www.wi-fi.org/beacon/alex-roytblat/fcc-must-proceed-on-6-ghz-nprm-to-meet-growing-spectrum-demand.

[5] H. Baldwin, "Wireless bandwidth: Are we running out of room?," January 2012. [Online]. Available: https://www.computerworld.com/article/2500312/mobile-wireless/wireless-bandwidth--are-we-running-out-of-room-.html.

[6] M. Oh, "A flicker mitigation modulation scheme for visible light communications," in *15th ICACT*, PyeongChang, 2013.

[7] B. Love, D. Love and J. Krogmeier, "Like deck chairs on the titanic: Why spectrum reallocation won't avert the coming data crunch but technology might keep the wireless industry afloat," *Washington University Law Review,* vol. 89, p. 705, 2012.

[8] D. Karunatilaka, F. Zafar, V. Kalavally and R. Parthiban, "LED Based Indoor Visible Light Communications: State of the Art," *IEEE Communications Surveys & Tutorials,* vol. 17, no. 3, pp. 1649-1677, 2015.

[9] H. Parikh, J. Chokshi, N. Gala and T. Biradar, "Wirelessly transmitting a grayscale image using visible light," in *ICATE*, Mumbai, 2013.

[10] I. O. f. Standardization, *Information technology -- Automatic identification and data capture techniques -- QR Code bar code symbology specification (ISO/IEC 18004:2015),* 2015.

[11] A. Lerner, A. Saxena, K. Ouimet, B. Turley, A. Vance, T. Kohno and F. Roesner, "Analyzing the Use of Quick Response Codes in the Wild," in *MobiSys '15*, New York, 2015.

[12] DENSO WAVE INCORPORATED, "History of QR Code," [Online]. Available: http://www.qrcode.com/en/history/.

[13] F. Miramirkhani, O. Narmanlioglu, M. Uysal and E. Panayirci, "A Mobile Channel Model for VLC and Application to Adaptive System Design," *IEEE Communications Letters,* vol. 21, no. 5, pp. 1035-1038, 2017.

[14] S. Arai, S. Mase, T. Yamazato, T. Endo, T. Fujii, M. Tanimoto, K. Kidono, Y. Kimura and Y. Ninomiya, "Experiment on Hierarchical Transmission Scheme for Visible Light Communication using LED Traffic Light and High-Speed Camera," in *66th Vehicular Technology Confrerence*, Baltimore, 2007.

[15] T. Nagura, T. Yamazato, M. Katayama, T. Yendo, T. Fujii and H. Okada, "Improved Decoding Methods of Visible Light Communication System for ITS Using LED Array and High-Speed Camera," in *71st Vehicular Technology Conference*, Taipei, 2010.

[16] F. Hermans, F. McNamara, G. Sörös, C. Rohner, T. Voigt and E. Ngai, "Focus: Robust Visual Codes for Everyone," in *MobiSys '16*, Singapore, 2016.

[17] S. Perli, *Pixnet: Designing Interference-free Wireless Links using LCD-Camera Pairs,* Massachusetts Institute of Technology, 2010.

[18] P. Ryan, "Application of the Public-Trust Doctrine and Principles of Natural Resource Management to Electromagnetic Spectrum," *Michigan Telecommunications and Technology Law Review,* vol. 10, no. 2, p. 285, 2004.

[19] A. G. Bell, "Apparatus for signaling and communicating, called photophone". US Patent 235199, 1880.

[20] N. Saha, I. Shareef, T. L. Nam and M. J. Yeong, "Survey on optical camera communications: challenges and opportunities," *IET Optoelectronics,* vol. 9, no. 5, pp. 172-183, 2015.

[21] F. R. Gfeller and U. Bapst, "Wireless in-house data communication via diffuse infrared radiation," *Proceedings of the IEEE,* vol. 67, no. 11, pp. 1474-1486, 1979.

[22] J. M. Kahn and J. R. Barry, "Wireless infrared communications," *Proceedings of the IEEE,* vol. 85, no. 2, pp. 265-298, 1997.

[23] "Infrared data association," [Online]. Available: http://www.irda.org/.

[24] P. H. Pathak, X. Feng, P. Hu and P. Mohaparta, "Visible Light Communication, Networking, and Sensing: A Survey, Potential and Challenges," *IEEE Communications Surveys & Tutorials,* vol. 17, no. 4, pp. 2047-2077, 2015.

[25] G. Pang, K.-L. Ho, T. Kwan and E. Yang, "Visible light communication for audio systems," *IEEE Transactions on Consumer Electronics,* vol. 45, no. 4, pp. 1112-1118, 1999.

[26] Y. Tanaka, S. Haruyama and M. Nakagawa, "Wireless optical transmissions with white colored led for wireless home links," in *PIMRC 2000*, London, 2000.

[27] VLCA, [Online]. Available: http://vlca.net/standard/.

[28] IEEE Std. 802.15.7, *IEEE Standard for Local and Metropolitan Area Networks-Part 15.7: Short-Range Wireless Optical Communication Using Visible Light,* IEEE, 2011.

[29] M. Uysal, F. Miramirkhani, O. Narmanlioglu, T. Baykas and E. Panayirci, "IEEE 802.15.7r1 Reference Channel Models for Visible Light Communications," *IEEE Communications Magazine,* vol. 55, no. 1, pp. 212-217, 2017.

[30] G. Woo, A. Lippman and R. Raskar, "VRCodes: Unobtrusive and active visual codes for interaction by exploiting rolling shutter," in *ISMAR*, Washington, DC, 2012.

[31] J. Ferrandiz-Lahuerta, D. Camps-Mur and J. Paradells-Aspas, "A Reliable Asynchronous Protocol for VLC Communications Based on the Rolling Shutter Effect," in *GLOBECOM*, San Diego, 2015.

[32] Y. Kuo, P. Pannuto, K. Hsiao and P. Dutta, "Luxapose: Indoor Positioning with Mobile Phones and Visible Light," in *MobiCom '14*, Maui, 2014.

[33] K. Kuraki, S. Nakagata, R. Tanaka and T. Anan, "Data transfer technology to enable communication between displays and smart devices," *FUJITSU Sci. Tech. J.,* vol. 50, no. 1, pp. 40-45, 2014.

[34] T. Li, C. An, T. Campbell and X. Zhou, "HiLight: Hiding bits in pixel translucency changes," *ACM SIGMOBILE Mobile Computing and Communications Review,* vol. 18, no. 3, pp. 62-70, 2015.

[35] A. Wang, Z. Li, G. Shen, G. Fang and B. Zeng, "InFrame++: Achieve simultaneous screen-human viewing and hidden screen-camera communication," in *MobiSys '15*, Florence, 2015.

[36] M. L. Montoya Freire and M. Di Francesco, "Reliable and bidirectional camera-based communications with smartphones," in *WOWMOM 2016*, Coimbra, 2016.

[37] W. Hu, H. Gu and Q. Pu, "LightSync: unsynchronized visual communication over screen-camera links," in *MobiCom '13*, Miami, 2013.

[38] H.-Y. Lee, *Unsynchronized visible light Communications using rolling shutter camera: implementation and evaluation,* M.S. thesis, Dept. of Comput. Sci. and Inform. Eng., College of Elect. Eng. and Comput. Sci., Nat. Taiwan University, 2014.

[39] F. Miramirkhani, M. Uysal, O. Narmanlioglu, M. Abdallah and K. Qaraqe, "Visible Light Channel Modeling for Gas Pipelines," *IEEE Photonics Journal,* vol. 10, no. 2, 2018.

[40] M. Querini and G. F. Italiano, "Color Classifiers for 2D Color Barcodes," in *Federated Conference on Computer Science and Information Systems*, 2013.

[41] A. Grillo, A. Lentini, M. Querini and G. F. Italiano, "High Capacity Colored Two Dimensional Codes," in *IMCSIT*, 2010.

[42] H. Oehlmann, "Method of reading a data carrier including multiple rows of bar code". US Patent 5235172, September 1992.

[43] Adams Communications, "Specifications For Popular 2D Bar Codes," [Online]. Available: http://www.adams1.com/stack.html.

[44] R. Want, B. N. Schilit and S. Jenson, "Enabling the internet of things," *Computer,* vol. 48, no. 1, pp. 28-35, 2015.

[45] Microsoft, "High Capacity Color Barcodes (HCCB)," December 2007. [Online]. Available: https://www.microsoft.com/en-us/research/project/high-capacity-color-barcodes-hccb/.

[46] I. Amidror, The Theory of the Moiré Phenomenon: Volume I: Periodic Layers, Springer, 2009.

[47] A. V. Oppenheim, A. S. Willsky and S. H. Nawab, Signals and Systems, Prentice-Hall, 1996.

[48] D. Parikh and G. Jancke, "Localization and segmentation of a 2d high capacity color barcode," in *2008 IEEE Workshop on Applications of Computer Vision*, 2008.

[49] O. Bulan, V. Monga and G. Sharma, "High capacity color barcodes using dot orientation and color separability," in *Media Forensics and Security*, San Jose, 2009.

[50] K. O. Siong, D. Chai and K. T. Tan, "The use of border in colour 2d barcode," in *ISPA 2008*, Sydney, 2008.

[51] T. Hao, R. Zhou and G. Xing, "COBRA: Color Barcode Streaming for Smartphone Systems," in *MobiSys '12*, Low Wood Bay, 2012.

[52] A. Wang, S. Ma, C. Hu, J. Huai, C. Peng and G. Shen, "Enhancing Reliability to Boost the Throughput over Screen-Camera Links," in *MobiCom '14*, Maui, 2014.

[53] R. Likamwa, D. Ramirez and J. Holloway, "Styrofoam: A Tightly Packed Coding Scheme for Camera-based Visible Light Communication," in *VLCS 2014*, Maui, 2014.

[54] M. Zhou, T. Lei, J. Li, Q. Wang, K. Ren and Z. Wang, "Rain Bar: Robust Application-Driven Visual Communication Using Color Barcodes," in *ICDCS*, Columbus, 2015.

[55] M. Xie, L. Hao, K. Yoshigoe and J. Bian, "CamTalk: A Bidirectional Light Communications Framework for Secure Communications on Smartphones," in *SecureComm 2013*, 2013.

[56] S. Hranilovic and F. R. Kschischang, "A pixelated MIMO wireless optical communication system," *IEEE Journal of Selected Topics in Quantum Electronics,* vol. 12, no. 4, pp. 859-874, 2006.

[57] S. Kim, X. Cao, H. Zhang and D. Tan, "Enabling concurrent dual views on common LCD screens," in *SIGCHI Conference on Human Factors in Computing Systems*, Austin, 2012.

[58] H. Liu, B. Liu, C. Shi and Y. Chen, "Secret Key Distribution Leveraging Color Shift Over Visible Light Channel," in *IEEE Conference on Communications and Network Security (CNS)*, Las Vegas, 2017.

[59] Apple Inc., "Use Night Shift on your iPhone, iPad, and iPod touch," [Online]. Available: https://support.apple.com/en-us/HT207570.

[60] W. Yuan, K. J. Dana, A. Ashok, M. Gruteser and N. Mandayam, "Spatially varying radiometric calibration for camera-display messaging," in *2013 IEEE Global Conference on Signal and Information Processing*, Austin, 2013.

[61] S. Daly, "Bit-depth extension of digital displays using noise". US Patent 6441867 B1, August 2002.

[62] S. Daly and X. Feng, "Bit-depth extension using spatiotemporal microdither based on models of the equivalent input noise of the visual system," in *Electronic Imaging 2003*, Santa Clara, 2003.

[63] S.-W. Lee and H. Nam, "A new dithering algorithm for higher image quality of liquid crystal displays," *IEEE Transactions on Consumer Electronics,* vol. 55, no. 4, pp. 2134-2138, 2009.

[64] R. Ulichney, Digital Halftoning, The MIT Press, 1987.

[65] E. Oat, M. Di Francesco and T. Aura, "MoCHA: Augmenting pervasive displays through mobile devices and web-based technologies," in *2014 IEEE International Conference on Pervasive Computing and Communication Workshops*, 2014.

[66] H. El Hattab, "reveal.js - The HTML Presentation Framework," [Online]. Available: https://revealjs.com/.

[67] "Socket.IO," [Online]. Available: https://socket.io.